

Google Ads Connector Template and General Guidelines

Table of Contents

GOOGLE ADS CONNECTOR TEMPLATE AND GENERAL GUIDELINES.....	1
TABLE OF CONTENTS.....	1
INFORMATICA CLOUD GOOGLE ADS CONNECTOR USER GUIDE	2
CHAPTER 1 - INTRODUCTION TO GOOGLE ADS CONNECTOR.....	2
GOOGLE ADS CONNECTOR OVERVIEW (REQUIRED).....	2
GOOGLE ADS CONNECTOR TASK AND OBJECT TYPES (REQUIRED).....	2
GOOGLE ADS OPERATIONS (BASED ON CONTENT)	2
INTRODUCTION TO GOOGLE ADS	5
Steps to be follow to generate the credential (Client_Id, client_secret and refresh_token) for Google Ads.....	5
ADMINISTRATION OF GOOGLE ADS CONNECTOR (REQUIRED).....	5
CHAPTER 2 - GOOGLE ADS CONNECTIONS	16
GOOGLE ADS CONNECTION OVERVIEW.....	16
GOOGLE ADS CONNECTION PROPERTIES	16
CHAPTER 3 - MAPPING AND MAPPING TASKS WITH GOOGLE ADS	17
GOOGLE ADS SOURCES IN MAPPING TASKS	17
READ OPERATION MAPPING	20
APPENDIX: DATA TYPE REFERENCE	22
DATA TYPE REFERENCE OVERVIEW.....	22
GOOGLE ADS AND TRANSFORMATION DATA TYPES	22
GUIDELINES FOR INFORMATICA INTELLIGENT CLOUD SERVICES/ INFORMATICA CLOUD DATA INTEGRATION DOCUMENTATION	23
PRODUCT NAMING CONVENTIONS	23
WHEN TO USE IICS AND CLOUD DATA INTEGRATION	23
TASKS NAMING CONVENTIONS.....	23
CONTENT, PROCEDURES, EXAMPLES, AND SCREENSHOTS.....	23
STYLE COMMENTS	23
WORDING.....	23
RELEASE NOTES GUIDELINES.....	24
MESSAGE WRITING EXAMPLES	28

INFORMATICA CLOUD GOOGLE ADS CONNECTOR USER GUIDE

Chapter 1 - Introduction to Google Ads Connector

Google Ads Connector Overview (Required)

You can use Google Ads Connector to read the different reports from Google Ads. It can be used for read operations for any report, for example Account-budget, campaign, etc....

Google Ads Connector Task and Object Types (Required)

This Connector is flat record based and it supports Mapping Task.

The following table lists the Google Ads object types that you can include in Informatica Cloud tasks:

Task Type	Source	Target
Mapping	Yes	Yes

Google Ads Objects (Based on content)

You can work with the following types of Google Ads objects in Data Integration:

Standard objects

The standard Objects supported by Google Ads are mentioned in the below table.

Object Name	Read	Insert	Delete	Update
Accessible_bidding_strategy	Yes	NA	NA	NA
Account_budget	Yes	NA	NA	NA
Account_budget_proposal	Yes	NA	NA	NA
Account_link	Yes	NA	NA	NA
Ad_group	Yes	NA	NA	NA
Ad_group_ad	Yes	NA	NA	NA
Ad_group_ad_asset_view	Yes	NA	NA	NA
Ad_group_ad_label	Yes	NA	NA	NA
Ad_group_asset	Yes	NA	NA	NA
Ad_group_audience_view	Yes	NA	NA	NA
Ad_group_bid_modifier	Yes	NA	NA	NA
Ad_group_criterion	Yes	NA	NA	NA
Ad_group_criterion_label	Yes	NA	NA	NA
Ad_group_criterion_simulation	Yes	NA	NA	NA
Ad_group_extension_setting	Yes	NA	NA	NA
Ad_group_feed	Yes	NA	NA	NA
Ad_group_label	Yes	NA	NA	NA
Ad_group_simulation	Yes	NA	NA	NA
Ad_parameter	Yes	NA	NA	NA
Ad_schedule_view	Yes	NA	NA	NA
Age_range_view	Yes	NA	NA	NA
Asset	Yes	NA	NA	NA
Asset_field_type_view	Yes	NA	NA	NA
Bidding_strategy	Yes	NA	NA	NA

Bidding_strategy_simulation	Yes	NA	NA	NA
Billing_setup	Yes	NA	NA	NA
Call_view	Yes	NA	NA	NA
Campaign	Yes	NA	NA	NA
Campaign_asset	Yes	NA	NA	NA
Campaign_audience_view	Yes	NA	NA	NA
Campaign_bid_modifier	Yes	NA	NA	NA
Campaign_budget	Yes	NA	NA	NA
Campaign_criterion	Yes	NA	NA	NA
Campaign_criterion_simulation	Yes	NA	NA	NA
Campaign_draft	Yes	NA	NA	NA
Campaign_experiment	Yes	NA	NA	NA
Campaign_extension_setting	Yes	NA	NA	NA
Campaign_feed	Yes	NA	NA	NA
Campaign_label	Yes	NA	NA	NA
Campaign_shared_set	Yes	NA	NA	NA
Campaign_simulation	Yes	NA	NA	NA
Carrier_constant	Yes	NA	NA	NA
Change_event	Yes	NA	NA	NA
Change_status	Yes	NA	NA	NA
Click_view	Yes	NA	NA	NA
Combined_audience	Yes	NA	NA	NA
Conversion_action	Yes	NA	NA	NA
Conversion_custom_variable	Yes	NA	NA	NA
Currency_constant	Yes	NA	NA	NA
Custom_audience	Yes	NA	NA	NA
Custom_interest	Yes	NA	NA	NA
Customer	Yes	NA	NA	NA
Customer_asset	Yes	NA	NA	NA
Customer_client	Yes	NA	NA	NA
Customer_client_link	Yes	NA	NA	NA
Customer_extension_setting	Yes	NA	NA	NA
Customer_feed	Yes	NA	NA	NA
Customer_label	Yes	NA	NA	NA
Customer_manager_link	Yes	NA	NA	NA
Customer_negative_criterion	Yes	NA	NA	NA
Customer_user_access	Yes	NA	NA	NA
Customer_user_access_invitation	Yes	NA	NA	NA
Detail_placement_view	Yes	NA	NA	NA
Detailed_demographic	Yes	NA	NA	NA
Display_keyword_view	Yes	NA	NA	NA
Distance_view	Yes	NA	NA	NA
Domain_category	Yes	NA	NA	NA
Dynamic_search_ads_search_term_view	Yes	NA	NA	NA
Expanded_landing_page_view	Yes	NA	NA	NA
Extension_feed_item	Yes	NA	NA	NA
Feed	Yes	NA	NA	NA
Feed_item	Yes	NA	NA	NA
Feed_item_set	Yes	NA	NA	NA
Feed_item_set_link	Yes	NA	NA	NA
Feed_item_target	Yes	NA	NA	NA
Feed_mapping	Yes	NA	NA	NA
Feed_placeholder_view	Yes	NA	NA	NA

gender_view	Yes	NA	NA	NA
geo_target_constant	Yes	NA	NA	NA
geographic_view	Yes	NA	NA	NA
group_placement_view	Yes	NA	NA	NA
hotel_group_view	Yes	NA	NA	NA
hotel_performance_view	Yes	NA	NA	NA
income_range_view	Yes	NA	NA	NA
keyword_plan_ad_group	Yes	NA	NA	NA
Keyword_plan_ad_group_keyword	Yes	NA	NA	NA
Keyword_plan_campaign	Yes	NA	NA	NA
Keyword_plan_campaign_keyword	Yes	NA	NA	NA
Keyword_theme_constant	Yes	NA	NA	NA
Keyword_view	Yes	NA	NA	NA
Label	Yes	NA	NA	NA
Landing_page_view	Yes	NA	NA	NA
Language_constant	Yes	NA	NA	NA
Life_event	Yes	NA	NA	NA
Location_view	Yes	NA	NA	NA
Managed_placement_view	Yes	NA	NA	NA
Media_file	Yes	NA	NA	NA
Mobile_app_category_constant	Yes	NA	NA	NA
Mobile_device_constant	Yes	NA	NA	NA
Offline_user_data_job	Yes	NA	NA	NA
Operating_system_version_constant	Yes	NA	NA	NA
Paid_organic_search_term_view	Yes	NA	NA	NA
Parental_status_view	Yes	NA	NA	NA
Product_bidding_category_constant	Yes	NA	NA	NA
Product_group_view	Yes	NA	NA	NA
Recommendation	Yes	NA	NA	NA
Remarketing_action	Yes	NA	NA	NA
Search_term_view	Yes	NA	NA	NA
Shared_criterion	Yes	NA	NA	NA
Shared_set	Yes	NA	NA	NA
Shopping_performance_view	Yes	NA	NA	NA
Smart_campaign_search_term_view	Yes	NA	NA	NA
Smart_campaign_setting	Yes	NA	NA	NA
Third_party_app_analytics_link	Yes	NA	NA	NA
Topic_constant	Yes	NA	NA	NA
Topic_view	Yes	NA	NA	NA
User_interest	Yes	NA	NA	NA
User_list	Yes	NA	NA	NA
User_location_view	Yes	NA	NA	NA
Video	Yes	NA	NA	NA
Webpage_view	Yes	NA	NA	NA

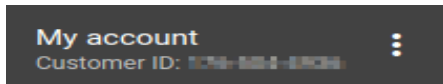
Introduction to Google Ads

The Google Ads is the modern programmatic interface to [Google Ads](#) and the next generation of the AdWords API. It enables developers to interact directly with the Google Ads platform, vastly increasing the efficiency of managing large or complex Google Ads accounts and campaigns.

Prerequisites

Developer Token : Developer token helps app to access the API. You can retrieve developer token by [signing in to your manager account](#) then navigating to the API Center (TOOLS & SETTINGS > SETUP > API Center).

Client customer ID : The client customer ID is the account number of the Google Ads client account you want to manage with the API, usually in the form **123-456-7890**.



OAuth2 client ID and client secret : The OAuth2 client ID and client secret map the client app to a project in the Google API Console and are used for OAuth2 authentication—allowing the app to operate on behalf of users account. They identify the app to Google and allow user to generate OAuth tokens for managing Google Ads users. (* **Note:** The word "client" here refers to app, not the Google Ads client account.)

OAuth2 access and refresh tokens : Before the app can access private data using the API, it must obtain an **OAuth2 access token** that grants access to the API. Follow the below steps for creating the refresh token.

Administration of Google Ads Connector (required)

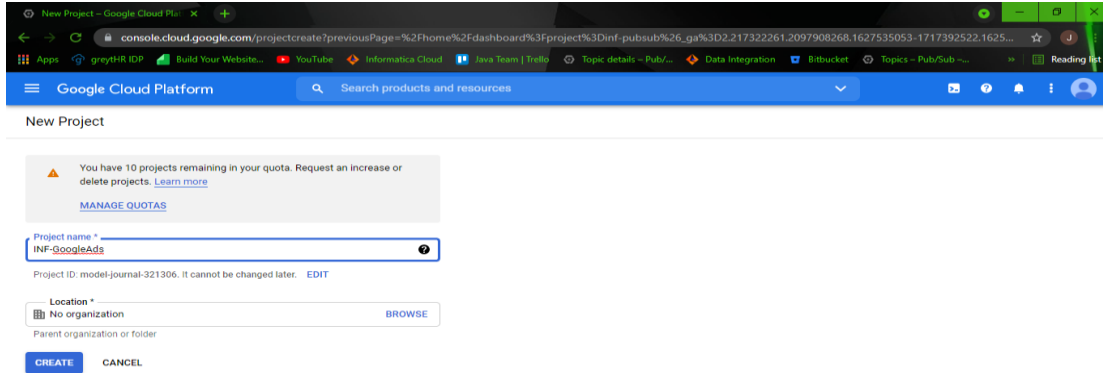
Steps to be follow to generate the credential (Client_Id, client_secret and refresh_token) for Google Ads

Setting up OAuth 2.0

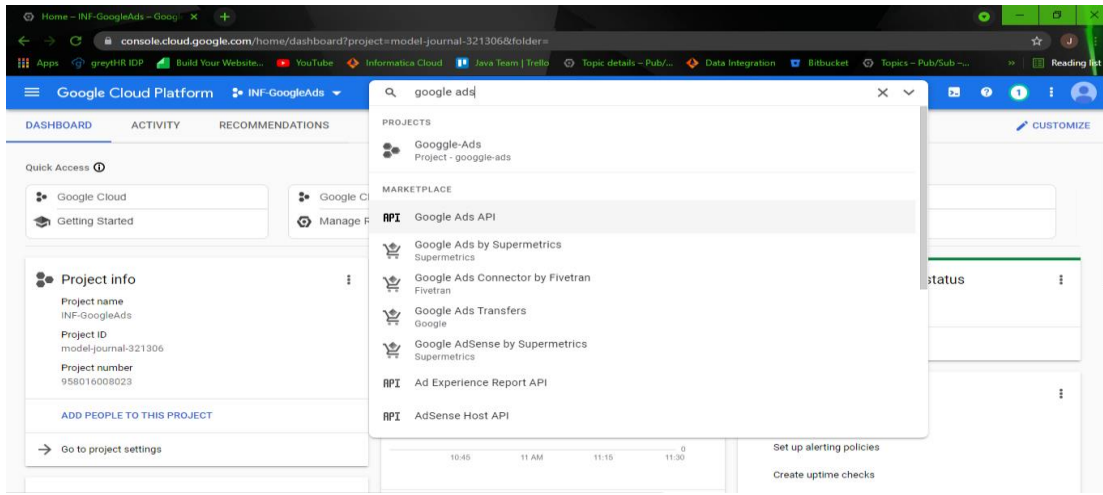
<https://support.google.com/cloud/answer/6158849?hl=en#userconsent&zippy=%2Cuser-consent>

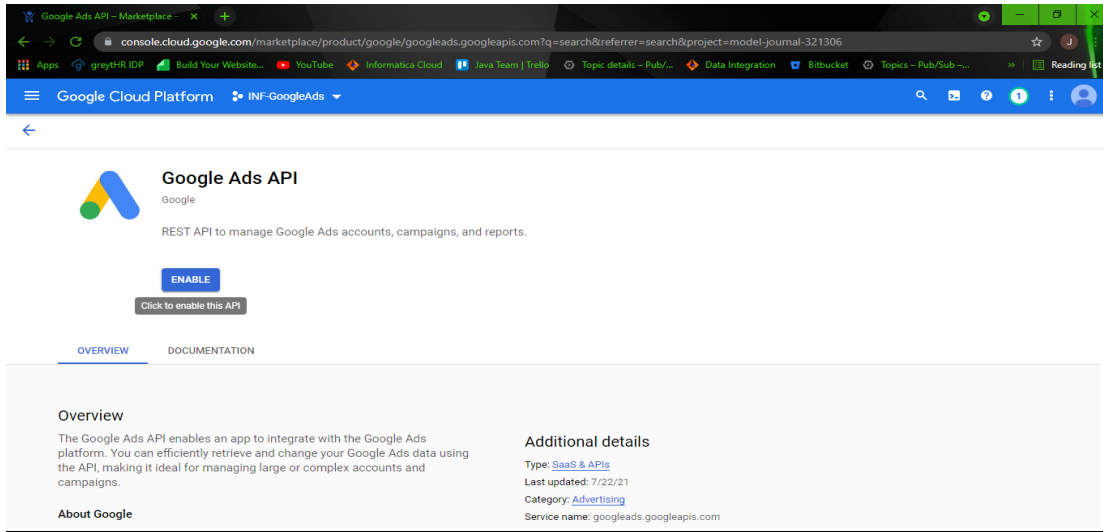
GCP : Google Cloud Platform

1. Create a new Project in GCP

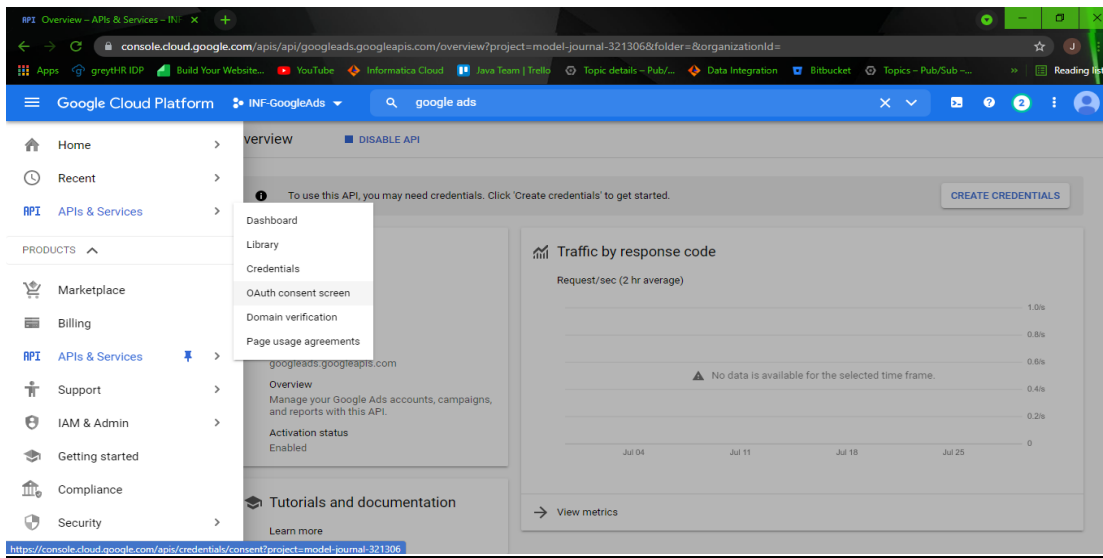


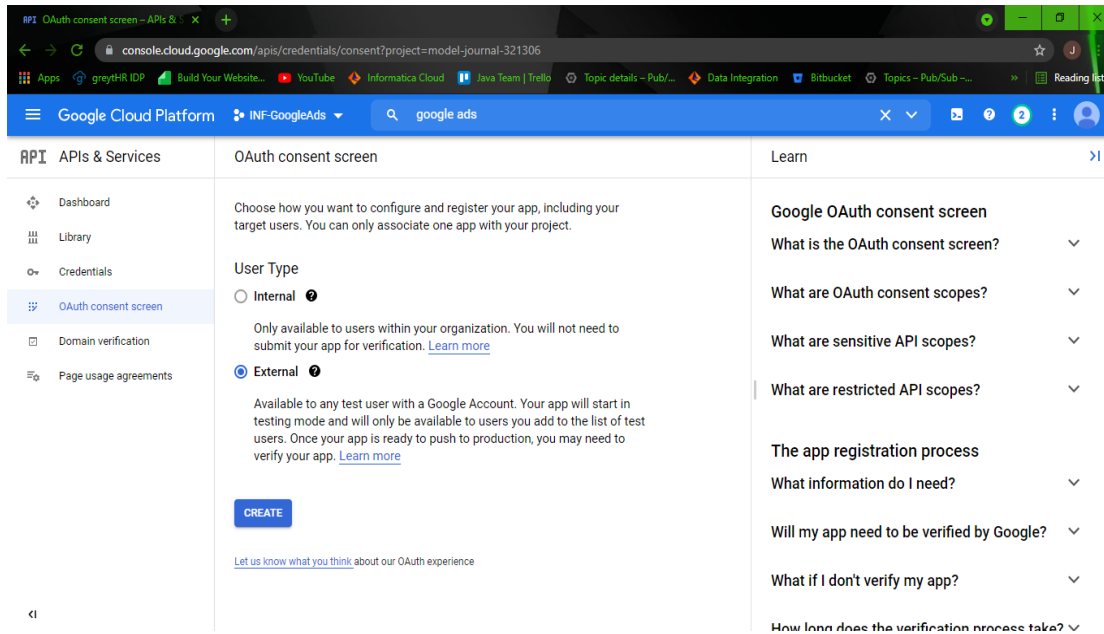
2. Enable the Google Ads API



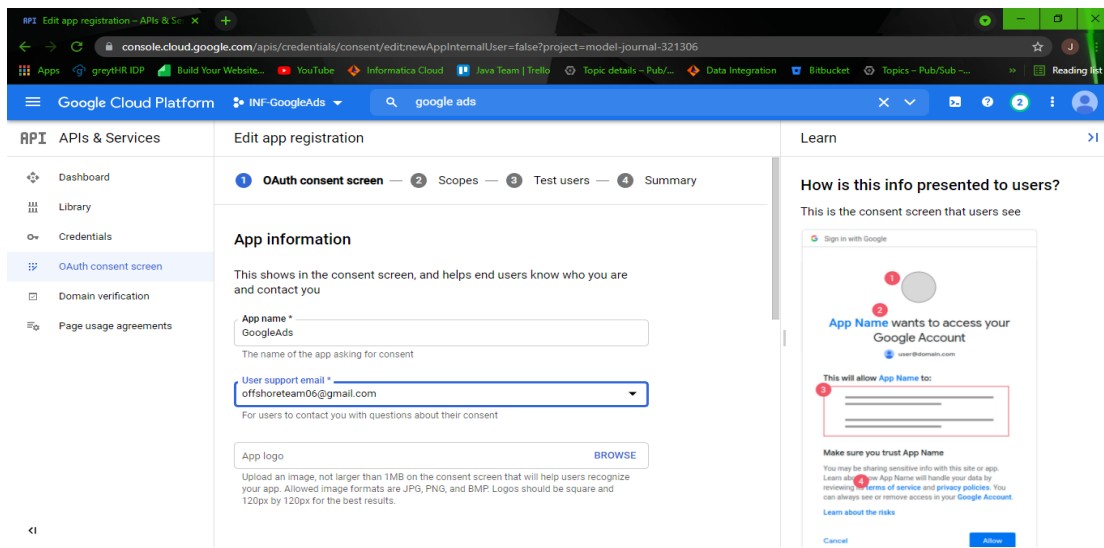


3. Now navigate to APIs and services and select “Oauth constant screen” and form User type select “External” and click on create.



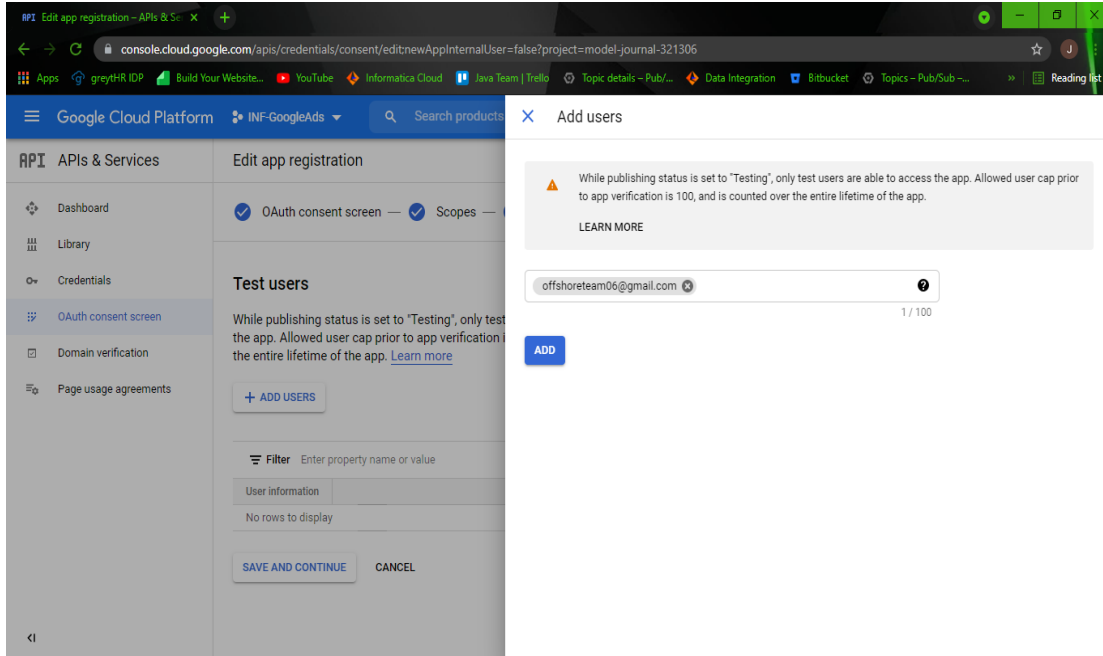


4. Provide the App name and, User support email (Same email using for GCP) and Developer Contact Information (Same email using for GCP).

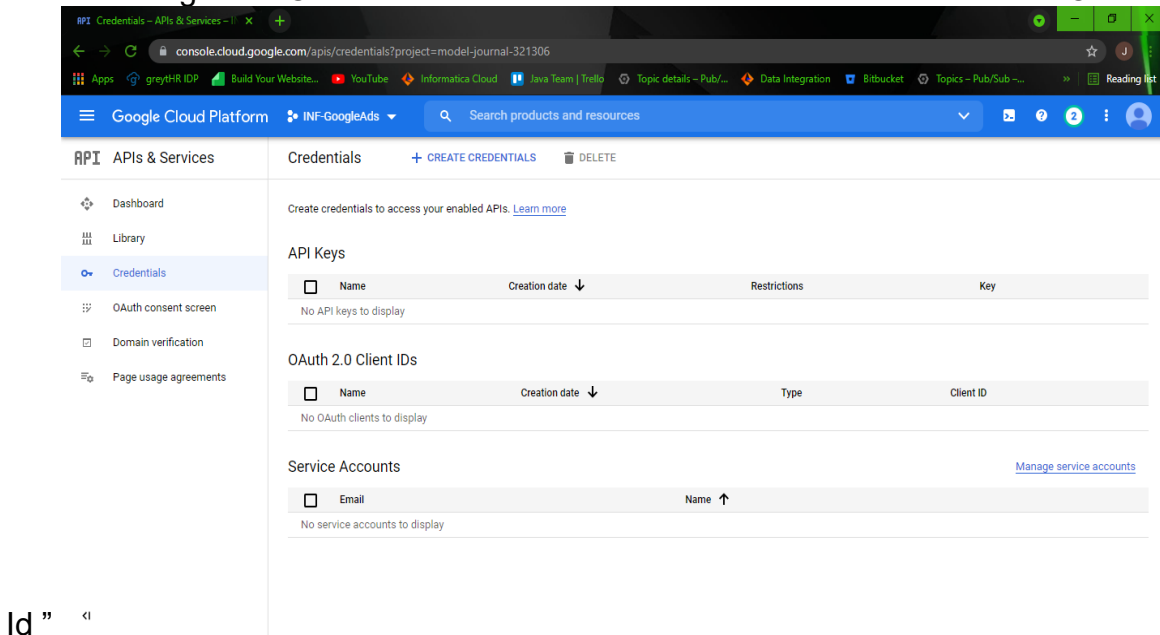


5. Click save and continue, no changes in Scope (save and continue).

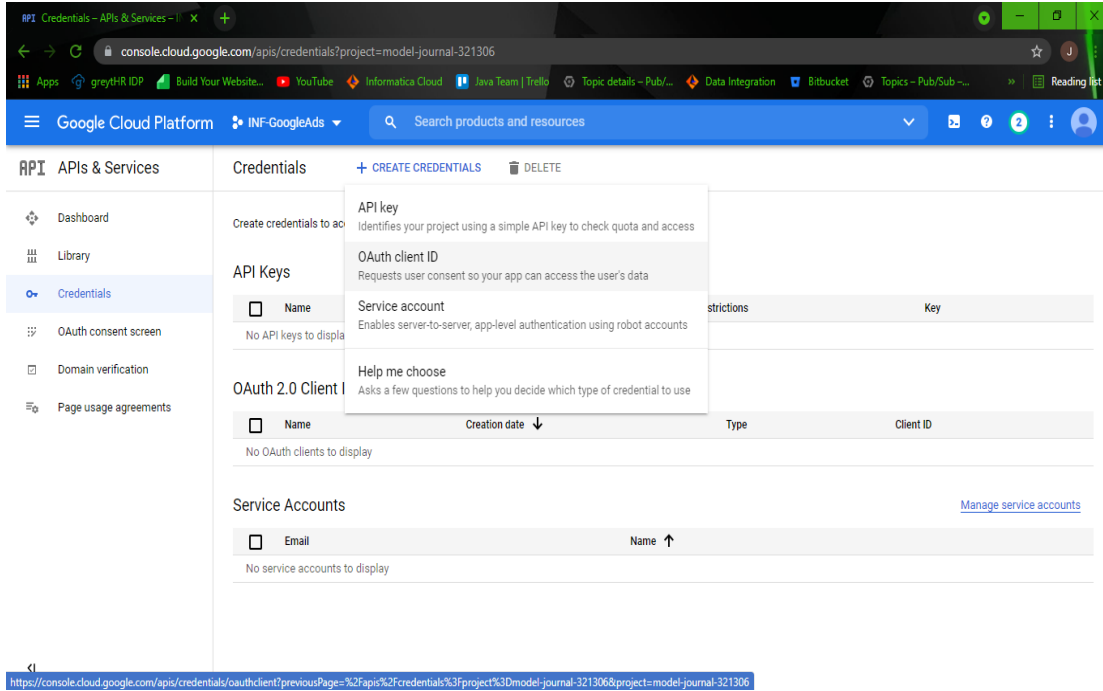
6. In Test users click on “add users” and provide the same email to add the test user and then save and continue.



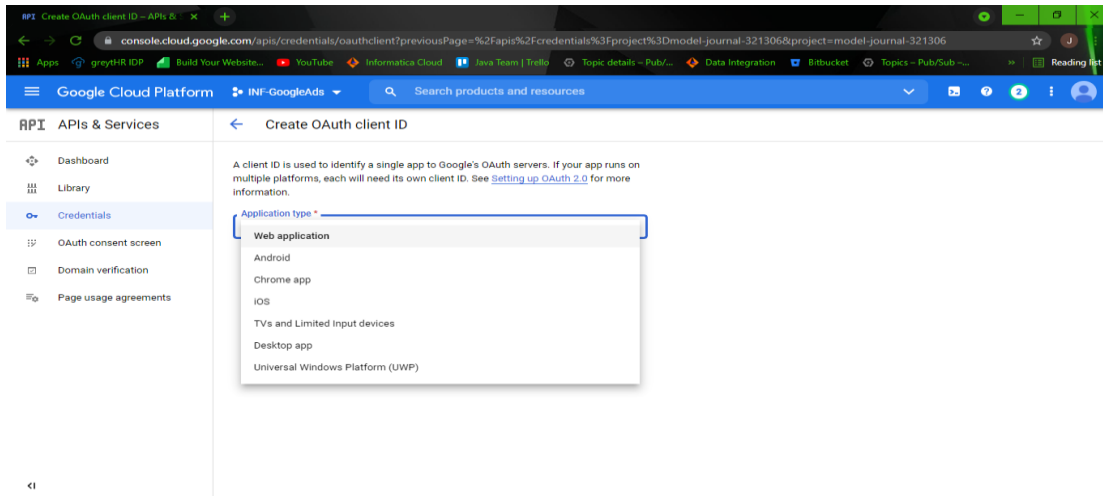
7. Now navigate to Credentials and click on “create credential” and select “OAuth client



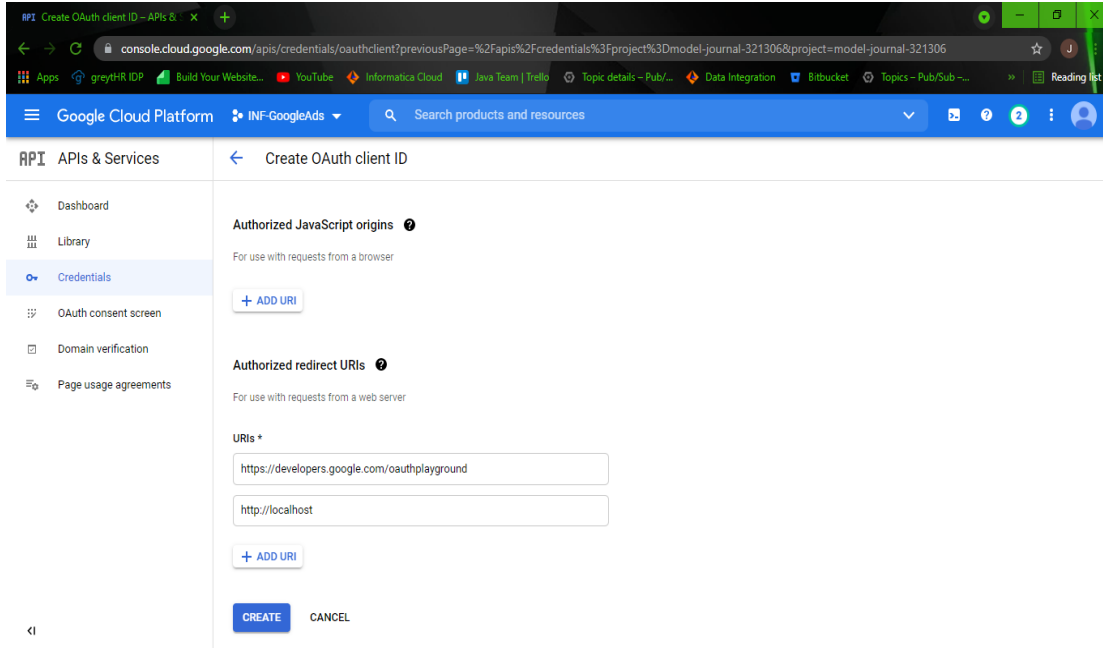
Id ” <



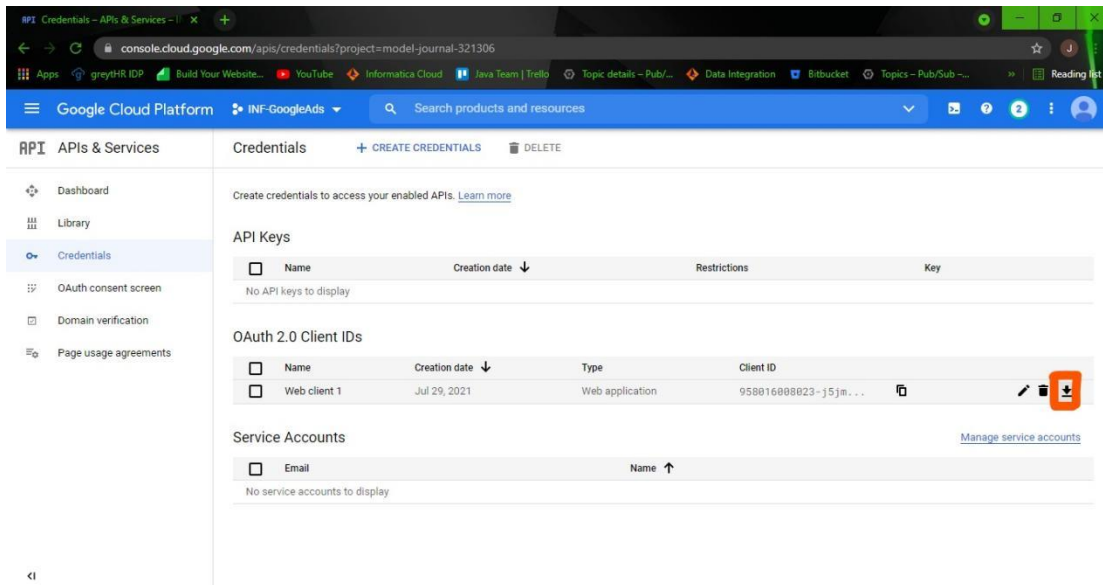
8. Select Application type as Web Application.



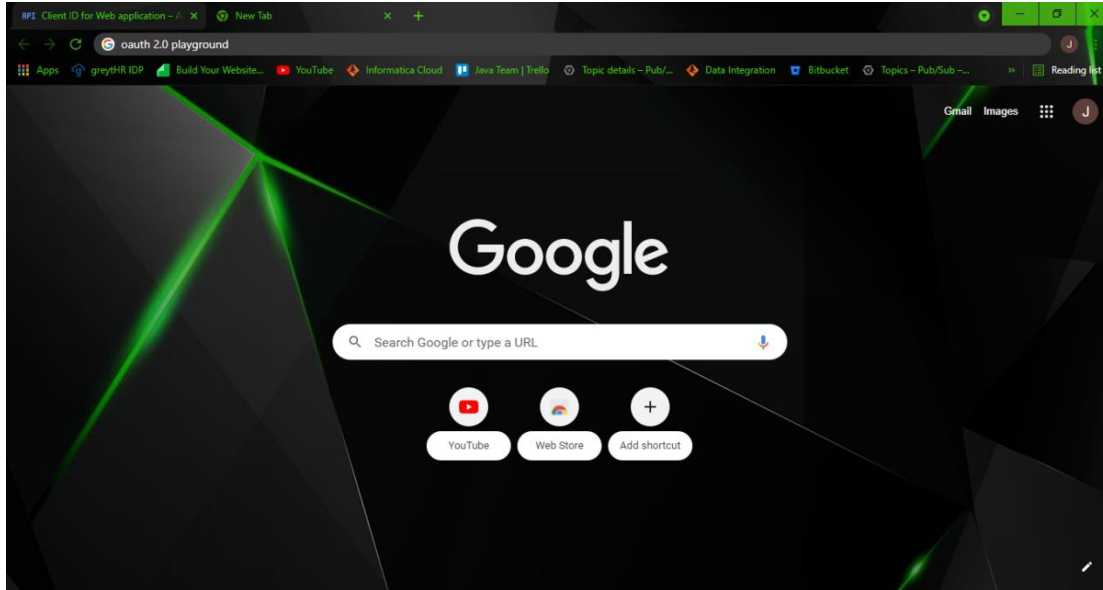
9. Provide the application name and add Authorized redirect uris.
- If Using OAuth 2.0 Playground, then to generate refresh token add <https://developers.google.com/oauthplayground>
 - If Using Postman, then to generate refresh token add : <http://localhost>



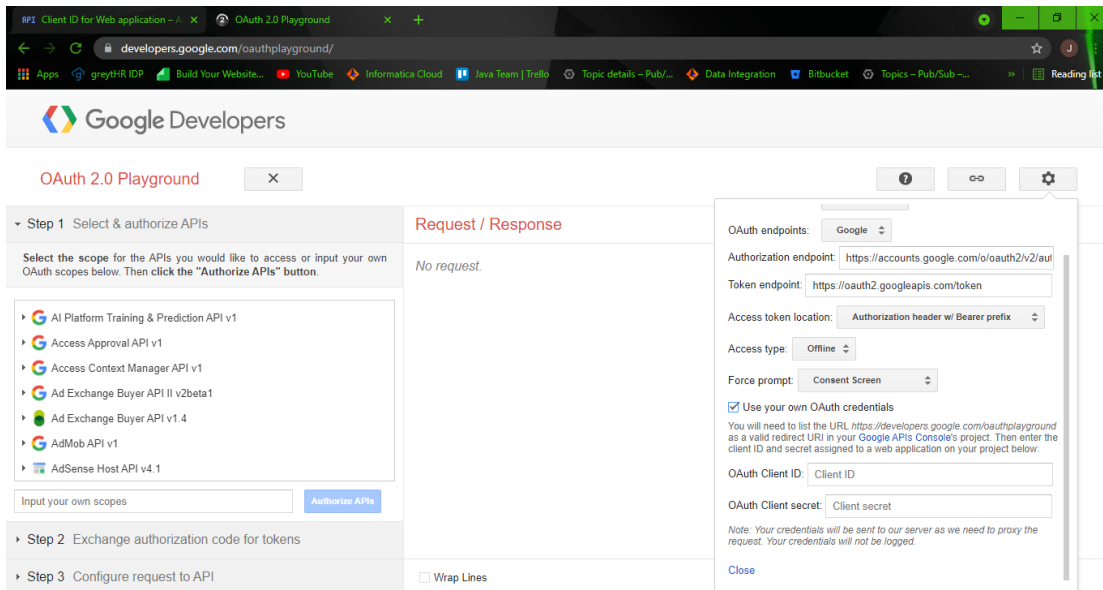
10. Download the created credentials which will contains Client_Id and client_secret.



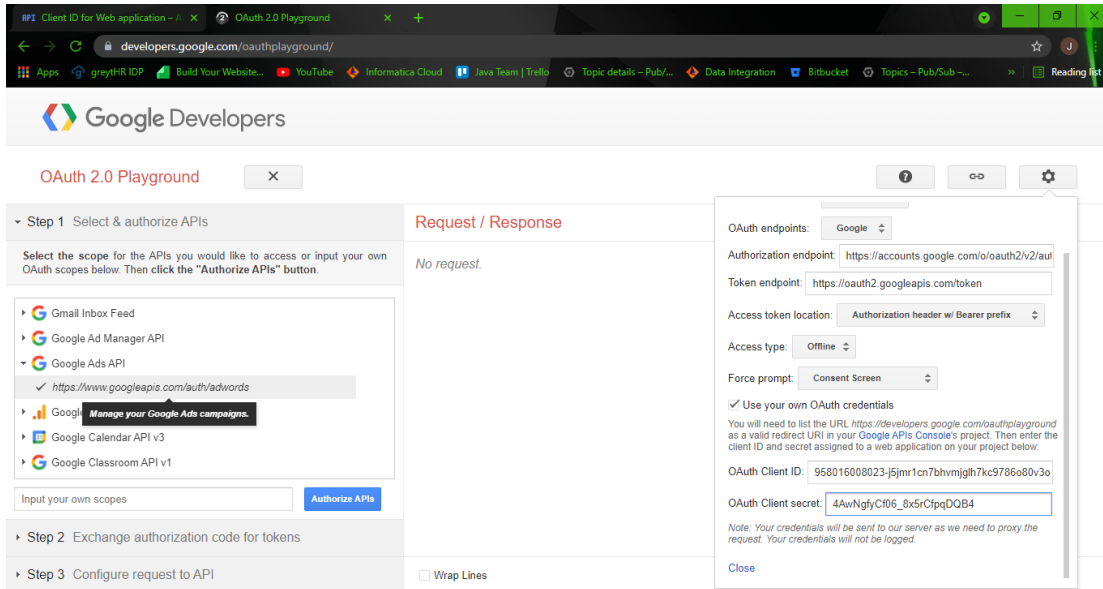
11. Search for OAuth 2.0 playground in browser.



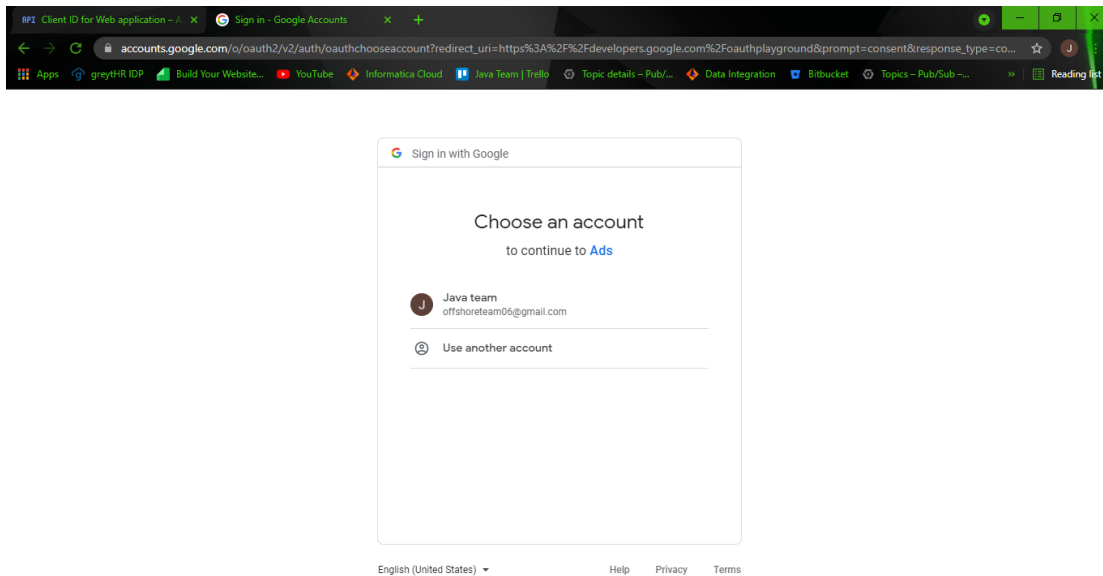
12. Click on setting icon and select “use your own OAuth credentials, and provide the Client_Id and client_secret which from GCP (Step 10)



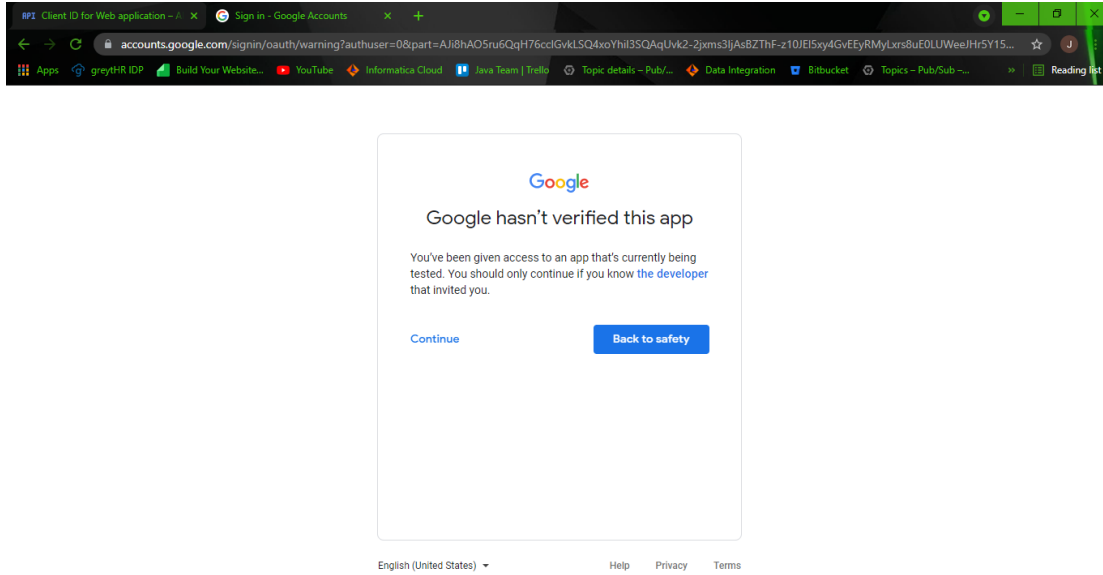
13. Expend the Google Ads and select <https://www.googleapis.com/auth/adwords> scope. Then click on Authorize APIs.



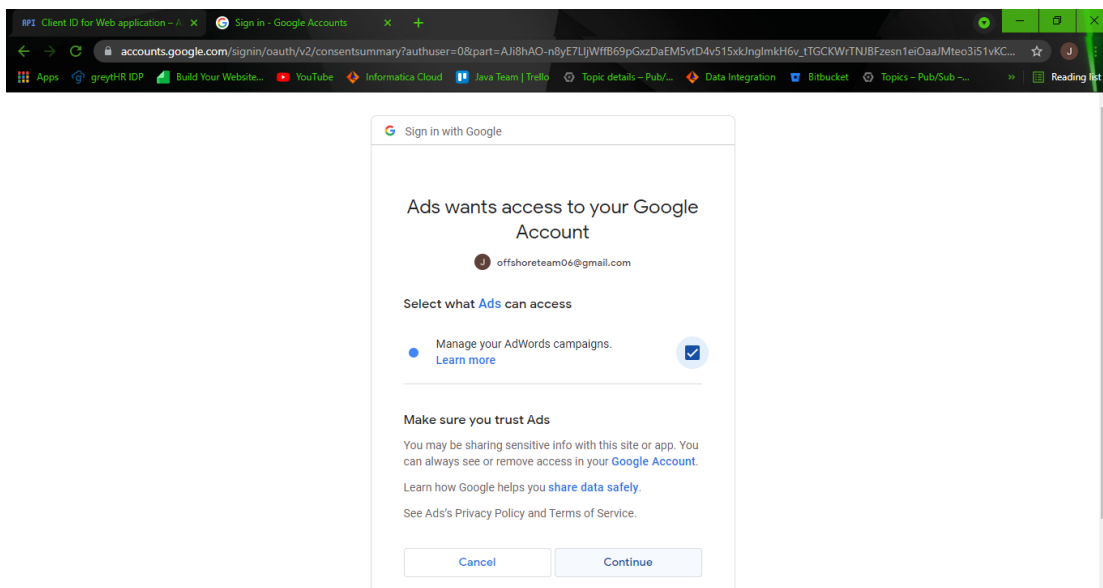
14. Choose an account which is using in GCP and click continue



15 Click on “Continue” button



16 Select the Check Box and again click on “Continue”



17 Click on “Exchange authorization code for tokens”

Step 2 Exchange authorization code for tokens

Once you got the Authorization Code from Step 1 click the **Exchange authorization code for tokens** button, you will get a refresh and an access token which is required to access OAuth protected resources.

Authorization code:

Exchange authorization code for tokens

Refresh token:

Access token: Refresh access token

Auto-refresh the token before it expires.

Note: The OAuth Playground will automatically revoke refresh tokens after 24h. You can avoid this by specifying your own application OAuth credentials using the Configuration panel.

Request / Response

```
HTTP/1.1 302 Found
Location: https://accounts.google.com/o/oauth2/v2/auth?
redirect_uri=https%3A%2F%2Fdevelopers.google.com%2Foauth2playground&prompt=consent&response_type=code&client_id=9580160
08023-
jsjwmc1cn7bhvsgj1h7kc9786o80v3o6f.apps.googleusercontent.com&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fadwords&ac
cess_type=offline

GET /oauth2playground/?code=4/0AX4XfWgM4YCoYnKY3KuHVN2RsDIzXajJEHvNDFN-XGRsOrGPTX-
rFxy3SQB8UWPgRscope=https://www.googleapis.com/auth/adwords HTTP/1.1
Host: developers.google.com
```

18 It'll generate the refresh token

Step 3 Configure request to API

Construct your HTTP request by specifying the URI, HTTP Method, headers, content type and request body. Then click the "Send the request" button to initiate the HTTP Request.

HTTP Method: Add headers

Request URI:

Enter request body Content-Type:

Send the request List possible operations

Note: The OAuth access token in Step 2 will be added to the Authorization header of the request.

Request / Response

```
HTTP/1.1 200 OK
Content-length: 415
X-xss-protection: 0
X-content-type-options: nosniff
Transfer-encoding: chunked
Expires: Mon, 01 Jan 1990 00:00:00 GMT
Vary: Origin, x-Origin, Referrer
Server: scaffolding on HTTPServer2
-content-encoding: gzip
-pragmas: no-cache
Cache-control: no-cache, no-store, max-age=0, must-revalidate
Date: Thu, 29 Jul 2021 07:44:18 GMT
X-frame-options: SAMEORIGIN
Alt-svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000,h3-T051=":443"; ma=2592000,h3-Q050=":443"; ma=2592000,h3-Q046=":443"; ma=2592000,h3-Q043=":443"; ma=2592000,quic=":443"; ma=2592000; v="46,43"
Content-type: application/json; charset=utf-8

{
  "access_token":
    "ya29.a0ARda9ddg9gziPjSCMhPuH8FXmo3PvKwTLIQUA09J05p5q_Ua51mTRHcaIHqXakYSkX_B235QINLoElGjoNhaIXADMV6cxpFaG4205P
    KRzUPpanSo7h4klyfIhgdl0V69cuJol-ZUBE-UJzGoKfAGV",
  "scope": "https://www.googleapis.com/auth/adwords",
  "token_type": "Bearer",
  "expires_in": 3599,
  "refresh_token":
    "1//04L911Ytb5Y3CpY2ARAAG03hw-L9I-JayzMSk1ep8idWQp0xe14YzFFxyf-
    KzeIQew8PYBZ51311aDzH01906K1MywVY"
}
```

Note : Use same gmail_Id for both GCP and Google Ads.

Chapter 2 - Google Ads Connections

Create a Google Ads connection to read data from Google Ads source and write data to a Google Ads target. You must create a connection for Google Ads Object that you want to connect. You can use Google Ads connections in mapping tasks.

Google Ads Connection Overview

Google Ads connector has 5 connection attributes, all of them are mandatory to create a Google Ads connection to access Google Ads data from Data Integration.

You can create a Google Ads connection on the Connections page. After you create a connection, it becomes available to all users who have access to the organization.

Google Ads Connection Properties

Connection Attributes	Description
Client ID	Specifies the clientID value present in the JSON file that you download downloaded from GCP.
Client Secret	Specifies the ClientSecret value present in the JSON file that you downloaded from GCP.
Refresh Token	Specifies the RefreshToken value that you generated using ClientID and ClientSecret in OAuth 2.0 Playground.
Developer Token	You will get developer token from your GoogleAds account.
Login CustomerID	You will get Login CustomerID from your GoogleAds account.

The following image gives the connection information:

Connection Details

Connection Name:*	<input type="text"/>
Description:	<input type="text"/>
Type: * ?	GoogleAds <input type="button" value="v"/>

GoogleAds Properties ?

Runtime Environment: * ?	Select... <input type="button" value="v"/>
--------------------------	--

Connection Section

CLIENT_ID: * ?	<input type="text"/>
CLIENT_SECRET: * ?	<input type="text"/>
REFRESH_TOKEN: * ?	<input type="text"/>
DEVELOPER_TOKEN: * ?	<input type="text"/>
LOGIN_CUSTOMER_ID: * ?	<input type="text"/>

Chapter 3 - Mapping and Mapping Tasks with Google Ads

Google Ads Sources in Mapping Tasks

accessible_bidding_strategy, account_budget, account_budget_proposal, account_link, ad_group, ad_group_ad, ad_group_ad_asset_view, ad_group_ad_label, ad_group_asset, ad_group_audience_view, ad_group_bid_modifier, ad_group_criterion, ad_group_criterion_label, ad_group_criterion_simulation, ad_group_extension_setting, ad_group_feed, ad_group_label, ad_group_simulation, ad_parameter, ad_schedule_view, age_range_view, asset, asset_field_type_view, bidding_strategy, bidding_strategy_simulation, billing_setup, call_view, campaign, campaign_asset, campaign_audience_view, campaign_bid_modifier, campaign_budget, campaign_criterion_simulation, campaign_draft, campaign_experiment, campaign_extension_setting, campaign_feed, campaign_label, campaign_shared_set, campaign_simulation, carrier_constant, change_event, change_status, click_view, combined_audience, conversion_action, conversion_custom_variable, currency_constant, custom_audience, custom_interest, customer, customer_asset, customer_client, customer_client_link, customer_extension_setting, customer_feed, customer_label, customer_manager_link, customer_negative_criterion, customer_user_access, customer_user_access_invitation, detail_placement_view, detailed_demographic, display_keyword_view, distance_view, domain_category, dynamic_search_ads_search_term_view, expanded_landing_page_view, extension_feed_item, feed, feed_item, feed_item_set, feed_item_set_link, feed_item_target, feed_mapping, feed_placeholder_view, gender_view, geo_target_constant, geographic_view, group_placement_view, hotel_group_view, hotel_performance_view, income_range_view, income_range_view, keyword_plan_ad_group, keyword_plan_ad_group_keyword, keyword_plan_campaign, keyword_plan_campaign_keyword,

keyword_theme_constant, keyword_view, label, landing_page_view, language_constant, life_event, location_view, managed_placement_view, media_file, mobile_app_category_constant, mobile_device_constant, offline_user_data_job, operating_system_version_constant, paid_organic_search_term_view, parental_status_view, product_bidding_category_constant, product_group_view, recommendation, remarketing_action, search_term_view, shared_criterion, shared_set, shopping_performance_view, smart_campaign_search_term_view, smart_campaign_setting, third_party_app_analytics_link, topic_constant, topic_view, user_interest, user_list, user_location_view, video, webpage_view, objects are supported under source.

Google Ads Target in Mapping Tasks

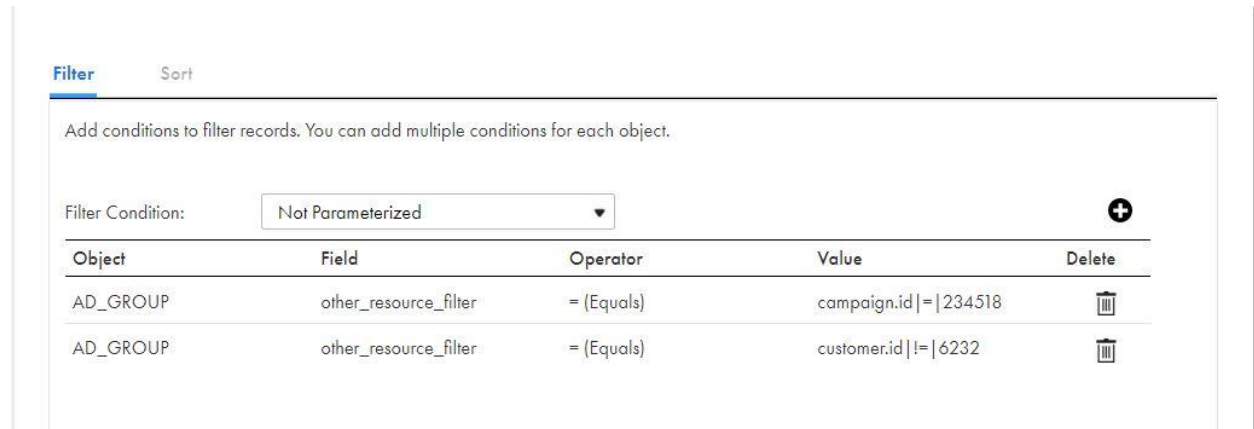
Not support any object under target.

Filter Fields:

In GoogleAds connector all fields from each object can be used as a filter fields. Every object contains one extra filter field with name "other_Resource_Filter". If user wants to give a filter for a field which is related to any other object, then other_Resource_Filter will be used.

For example in Ad_Group object if wants to use campaign_id as a filter then we have to use other_Resource_Filter with following format
 "object_Name.field_Name|operator(=, !+, <, >)|value "

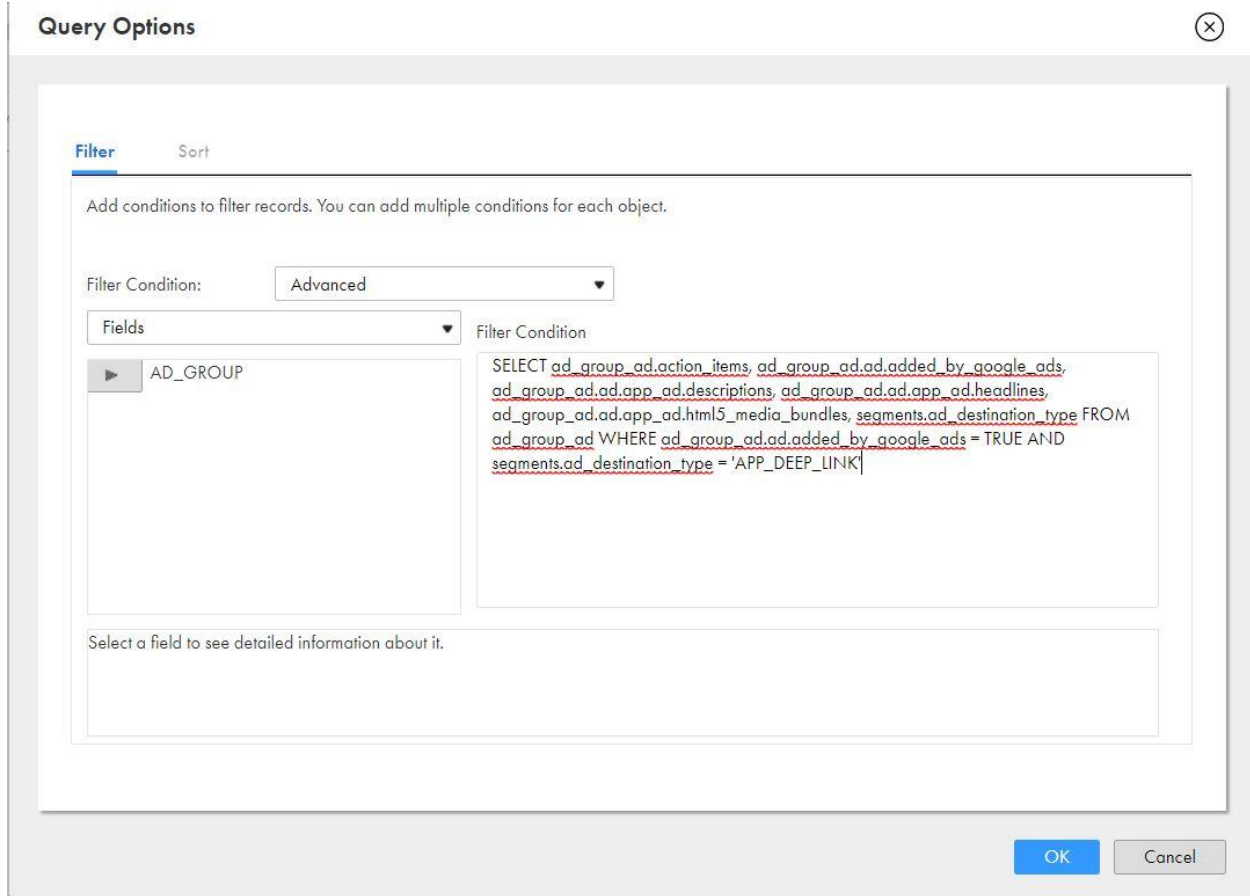
And also user can use any number of other_Resource_Filter.



Based on requirement user can also give their own GAQL (Google Ads Query Language) query through advanced filter.

To create a GAQL query use this URL

https://developers.google.com/google-ads/api/fields/v8/ad_group_ad_query_builder



Read Attributes:

Object	Attribute Name	Data Type
All Objects	customer_id	string
All Objects	OrderByClause	string
All Objects	limit	integer

Customer-id is a mandatory attribute for all objects and other two read attributes are optional user can use these two based on requirement.

Customer_id : Login CustomerID used in connection attribute is universal and unique for a google ads manager account. Read attribute's customer-id is an operating customer-id and a google ads manager account can have several operating customer-id.

OrderByClause : This attribute can be used if the user wants to get the data based on ascending or descending order. The format for this attribute is “object_Name.field_Name=ASC/DESC”.

User can give multiple values here through comma-separated for example

“ad_group.name=DESC,ad_Group.id=ASC”

Limit : Limit Attribute can be used to restrict the number of records.

Examples of Google Ads Mapping Tasks

Read Operation Mapping

1. In Mapping for Google Ads Read operation, specify Task name as **GoogleAds_Read**, select as source Google Ads connection, select Target and give Properties to Source and target.

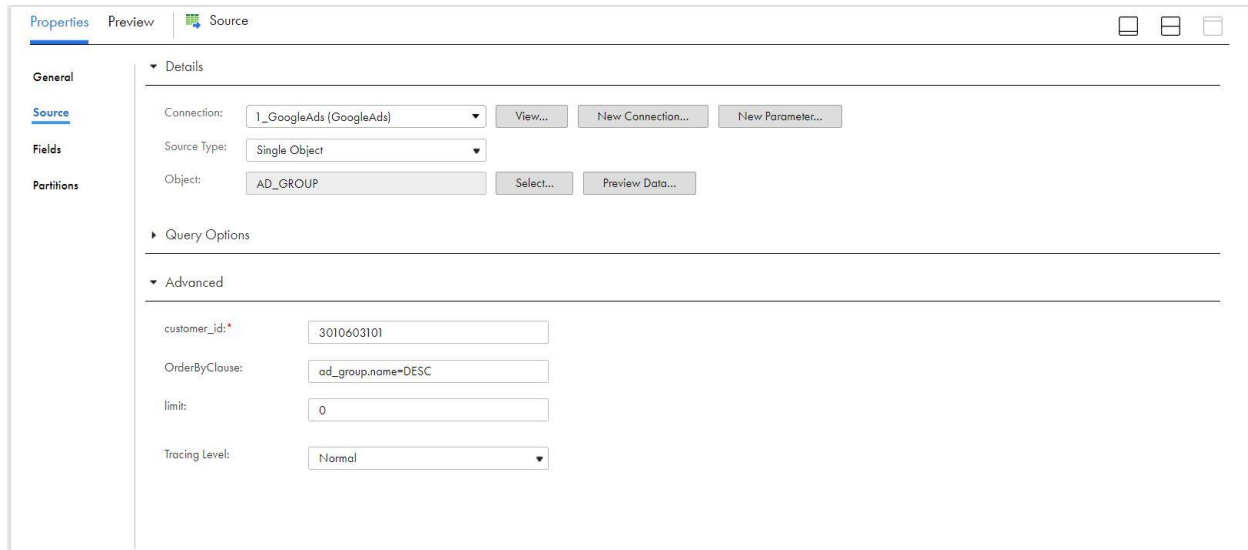
Select Source connection which you have created for Google Ads.

Source properties: -

Connection: GoogleAds

Source type: Single Object

Object: AD_Group



The screenshot shows the 'Properties' dialog box for a 'Source' connection. The 'Source' tab is selected. The 'Details' section contains the following fields:

- Connection: 1_GoogleAds (GoogleAds)
- Source Type: Single Object
- Object: AD_GROUP

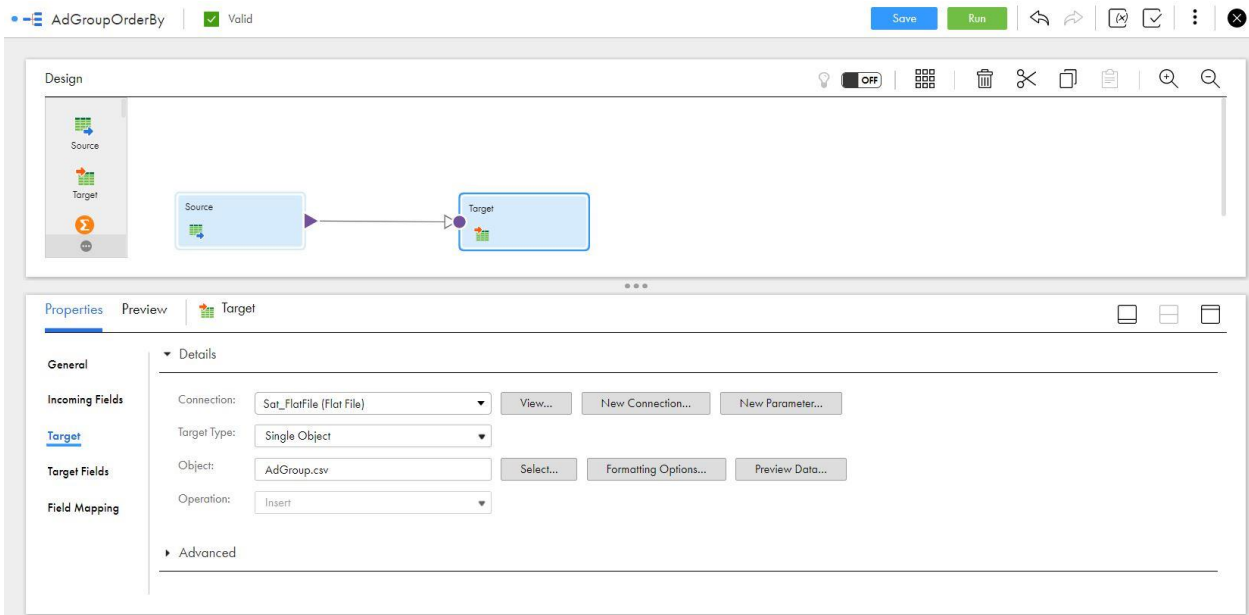
The 'Advanced' section contains the following fields:

- customer_id*: 3010603101
- OrderByClause: ad_group.name=DESC
- limit: 0
- Tracing Level: Normal

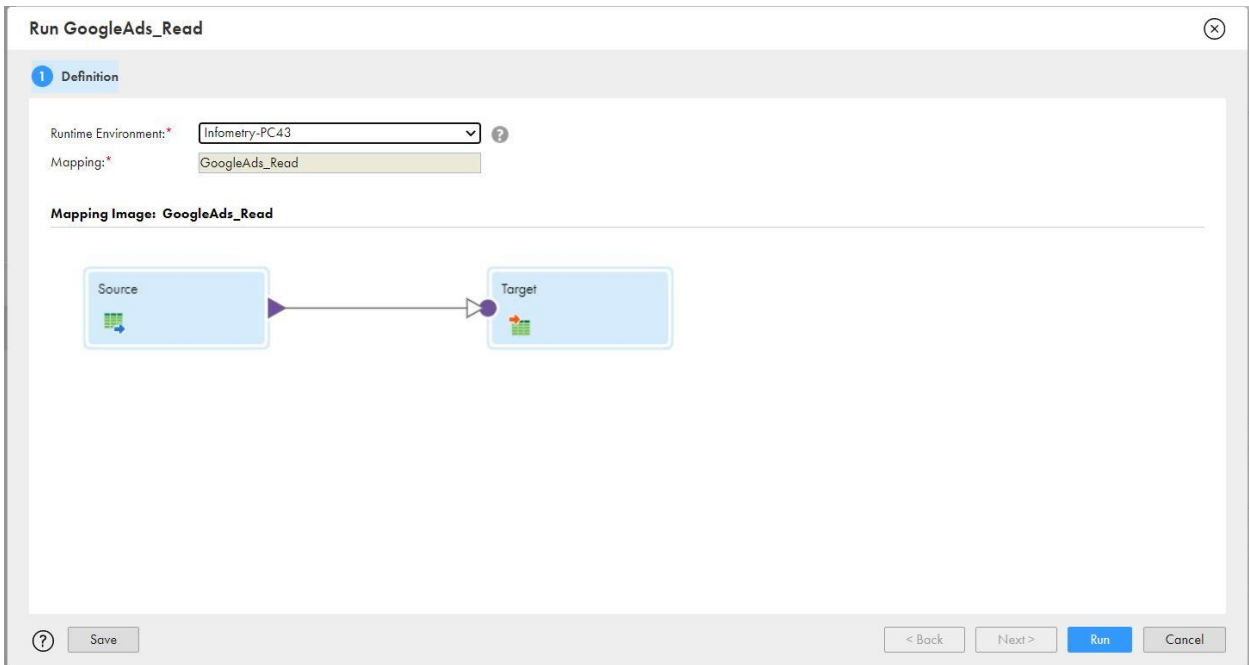
2. In Mapping for Google Ads Read operation, select target connection which you have created for Target.

Target properties: -

- a. Connection: FlatFile_Connection
- b. Target type: Single Object
- c. Object: AdGroup.csv
- d. Operation: Insert



3. In Mapping configuration task for Google Ads Read operation, specify Definition task name as **GoogleAds Read**, Select Runtime environment and run the task.



4. Click on My Jobs to see the result.

Appendix: Data Type Reference

Data Type Reference Overview

Data Integration uses the following data types in mappings and mapping tasks with GoogleAds

Google Ads data types:

Google Ads data types appear in the Fields tab for Source and Target transformations when you choose to edit metadata for the fields.

Transformation data types:

Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the Secure Agent uses to move data across platforms. Transformation data types appear in all transformations in a mapping.

When Data Integration reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When Data Integration writes to a target, it converts the transformation data types to the comparable native data types.

Google Ads and Transformation Data Types

The following table lists the supporting Google Ads data types and the corresponding transformation data types:

Google Ads Data Type	Transformation Data Type	Description
String	String	1 to 10485760 characters

GUIDELINES FOR INFORMATICA INTELLIGENT CLOUD SERVICES/ INFORMATICA CLOUD DATA INTEGRATION DOCUMENTATION

We need to maintain and create content for legacy Informatica Cloud and Informatica Intelligent Cloud Services (IICS) aka Florence concurrently.

Product Naming Conventions

The official name for Florence is **Informatica Intelligent Cloud Services (IICS)**. IICS is a parent product and includes multiple Informatica products (**Informatica Cloud**, ICRT, DQ, MDM, and so on). The official name for the new version of Informatica Cloud is **Informatica Cloud Data Integration** when used on IICS.

Product name in legacy connector user guides: **Informatica Cloud**

Product name in Florence connector user guides: **Informatica Cloud Data Integration**

Parent product name in Florence connector user guides: **Informatica Intelligent Cloud Services**

When to use IICS and Cloud Data Integration

Use **Informatica Intelligent Cloud Services** as the product name whenever you are talking about the platform. For example, a user logs in to IICS (not Cloud Data Integration), and you create IICS users (not Cloud Data Integration users).

Use Cloud Data Integration when referring to the Integration Designer only. For example, you create mappings, mapping tasks, and task flows in Cloud Data Integration.

Tasks Naming Conventions

The following table lists tasks names for Informatica Cloud and Cloud Data Integration:

Informatica Cloud	Cloud Data Integration
Data Synchronization task	synchronization task
Data Replication task	replication task
Mapping Configuration task	mapping task
Data Masking task	data masking task

Note: Write task names for Cloud Data Integration in lower case unless they are used in headings, as UI labels, or in the beginning of the sentences.

Content, Procedures, Examples, and Screenshots

Make sure that all the content including examples and screenshots have been modified as per the Cloud Data Integration UI in the Florence connector guides.

STYLE COMMENTS

Wording

Connector does not require an article. For example, write “Use Marketo REST Connector...” instead of “Use the Marketo REST Connector...”

Note wording for Secure Agent and runtime environments. Always use the content references that we have set up. This terminology has been through tech review, but the development is still evolving, and the terminology might change.

RELEASE NOTES GUIDELINES

Rules and Guidelines for New Features

When you document a new feature in a release guide, ensure that the content follows the standard structure.

Rules and guidelines for all release guides

Consider the following rules and guidelines:

- Use the following lead-in sentence for each version topic: "This section describes new features in version *<version>*."
- Alphabetize the sub-topics and sections in each functional category topic, unless you receive different guidance from product management. PM might occasionally want to highlight certain information first.
- In general, do not document new UI features. Document changed behavior, not changed appearance. For example, if there is a new dialog box for unlocking repository objects in Developer tool, add an entry about unlocking objects in the Developer tool. The exception is if the UI for the entire tool has changed.
- Optionally, add images to show UI or other changes when you think these will benefit the user.
- Include a book reference for each feature.

Use the following syntax for the book reference: "For more information, see the *<full book name, including version number>*." Use the *<cite>* tag for the product name, version, number, and book name. For example, "See the *Administrator Guide*."

- Optionally, include a chapter reference for a feature.

Known Limitations

A known limitation is a limitation of the designed and implemented product. Unsupported functionality is not a limitation. The known limitations section of the release notes provides brief descriptions and workarounds to high priority and high severity product limitations.

Important: Limitations can be difficult to write. Although you might write just a few sentences, you must be fully familiar with the product and the circumstances that surround the limitation before you can convey it accurately and succinctly in the Release Note writeup. Do not rely on terminology in the bug tracking system, as much of the terminology is internal. Verify that you use terminology that is consistent with user doc and the product.

Known Limitations Criteria

Before you document a known limitation, verify that the limitation meets criteria for the release notes. Informatica includes high priority and high severity product limitations that Development is committed to fixing.

When you review known limitations that are release note candidates, consider the following criteria:

Limitation type

Known limitations must be functional limitations. Do not include enhancement requests or documentation limitations. If the nature of the limitation is functional and the type is "Documentation," look for the linked functional limitation.

Content to include

After you verify the limitation type, priority, and status, review the content of the limitation. Include the following types of errors in the known limitations:

- Product failures, such as hanging, termination, or error response to user request
- System failures, such as core dumps and crashing
- Inconsistent data or unexpected results
- Regressions
- Any other issue that might result in a call to Informatica Global Customer Support

Content to exclude

Do not include the following types of bugs in known limitations:

- Any limitation that might affect install or upgrade. For example, the installation fails with memory errors, or the upgrade fails to upgrade privileges properly. Document all install and upgrade limitations in the Installation topic of the release notes.
- PAM-related issues of supported systems. For example, do not include a limitation requesting the support of a particular version of an OS for a product.
- Corner case bugs, or bugs that are unlikely to be found by a customer. If a limitation is extremely difficult to reproduce or was possibly discovered by QA under severe stress testing, consider exclusion from the release notes.
- Requests for additional functionality. A bug that indicates a request for support of additional functionality or a feature is not considered a limitation for release notes. Key words to watch for are "support" and "should have."
- Bugs that are minor irritations. This can include misaligned user interface text, misspellings, and too many clicks or scrolls.
- Bugs that are not reproducible.

Exceptions

If you question whether to include a limitation or not, consider whether excluding the limitation from the release notes would result in a call to GCS. Note the following circumstances when we can relax the release note criteria:

- New products with a customer base that is vocal about documenting all limitations
- End-user products where limitations such as abnormal scrolling or too many clicks are higher priority

- Special requests from product management, development, or QA

Known Limitations Write-up

Write up a known limitation to describe the limitation instead of the expected behavior. Include any workaround.

A known limitation has the following elements:

Bug number

The bug number is the functional bug number in Jira. If Development linked a bug for the doc impact, do not use the documentation bug number.

Bug description

When you write a known limitation, write a short description of the bug behavior instead of the expected behavior.

- **Change:** The RestoreDomain command should not generate an exception if you set the -tc option.
- **To:** The RestoreDomain command generates an exception if you set the -tc option.

Workaround

If a limitation has a workaround, include it under the limitation. If Jira does not include a workaround, ask Dev/QA to provide one.

Do not document the following types of bugs as limitations that have the following workarounds:

Custom property

Document custom properties, called undocumented flags, as an internal KB article.

System patch

Document operating system patches in the installation section of the release notes. QA provides this information to Documentation. If the workaround is a patch that is not in the list of patches, verify with QA whether it belongs in the list. If you need to document the behavior, it might fit as a separate topic in the installation section of the release notes.

Fixed Limitations

Document fixed limitations that were reported by customers or were reported as limitations in a previous release.

When you write a fixed limitation, write a short description of the bug behavior instead of the fixed behavior.

- **Change:** The Column Profiling Details dialog box appears when you view the column profile for a source column in a mapping specification.
- **To:** The Column Profiling Details dialog box does not appear when you view the column profile for a source column in a mapping specification.

If a fixed limitation was previously documented as a known limitation, move the entry to the Fixed topic. Delete any workaround that was documented with limitation.

Fixed Limitations Criteria

Before you write a fixed limitation, verify that it meets the criteria for the release notes.

Most of the time, bugs are fixed in the code, tested, and closed. However, Dev might close a bug for multiple reasons. For example, it might be a duplicate of another bug, or it might be closed with a workaround. When you review a closed bug, you need to read the closing comments at the end of the notes to find out why a bug was actually closed. If it is not checked in to the code, we cannot publish it as being closed.

When you review fixed limitations that are release note candidates, consider the following criteria:

Fixed release

Verify that the limitation was verified and closed in the release that you are documenting.

Note: If the limitation was linked from a previous release and this fix was merged from a previous release, do not document it again as fixed. The content reference informing customers about fixes in previous releases is sufficient.

Status

Verify that the status of the limitation is "closed." If it is "resolved," verify with Dev and QA that they will be able to close it for the release. Watch the bug for the status change.

Content to include

Include fixed limitations that meet the following conditions:

- The fix was checked in to the code.
- The bug was not opened in the current release.
- The bug was previously documented as a known limitation, or it was reported by a customer or GCS.

Content to exclude

Do not include fixed limitations that meet the following conditions:

- If a limitation is closed as a duplicate, look at the linked bug to see if it belongs with known limitations.
- If a limitation is closed as "will not fix" or "as-designed," consider documenting the issue in the Knowledge Base or user documentation. If the behavior still seems buggy, put this in a Knowledge Base article. If the workaround, or the user actions required to get the desired behavior are fairly simple, consider including it in the user docs.
- If a limitation is closed with a workaround, you can include it in a Knowledge Base article. Workarounds can include undocumented flags, registry edits, and system patches.

- If an issue was issue that is closed with a custom property, it is a candidate for the Knowledge Base. We do not expose these properties to all customers, so any documentation will be through an internal KB article. Ask Global Customer Support and QA if they want this documented.
- If a limitation is closed with a note that Documentation is adding it as a limitation in the release notes, let Development know that we cannot document anything in the release notes without Dev commitment to fix.

MESSAGE WRITING EXAMPLES

The examples in this section show messages that have been rewritten to follow the message writing guidelines.

The following table shows original and edited messages and the guidelines used for the rewrite:

Original Message	Edited Message	Guidelines for Rewrite
Attachment file {filename} not found.	Cannot find the post-session email attachment file {filename}.	- Provide as much information as possible.
Error executing stored procedure...	An error occurred while executing the stored procedure for transformation {transformationname}.	- End each message with a period. - Provide as much information as possible. - Qualify all parameters.
NULL external procedure name.	The External Procedure transformation {transformationname} failed because the name of the external procedure is null. Specify the name of the external procedure.	- Provide as much information as possible. - Qualify all parameters. - Tell where the error occurred.
Getting free block from exchange failed!!	Internal error. The Integration Service encountered a fatal error while getting a block from the buffer pool. Contact Informatica Global Customer Support.	- End each sentence with a period. - Provide as much information as possible. - Use words that the customer will understand.
ERROR: Field name used in join not found in transform definition.	Internal error. The Integration Service cannot find the port {portname} used in the join condition for the Joiner transformation {transformationname}. Contact Informatica Global Customer Support.	- Provide as much information as possible. - Tell where the error occurred. - Qualify all parameters. - Spell out words.
Pushdown optimization is not supported because you	Pushdown optimization is skipped because session	-

enabled row error logging. To use pushdown optimization, disable row error logging.

{sessionname} has row error logging enabled.

Avoid offending the user.

- Provide as much information as possible.

{transformationname} cannot be pushed to the target database because it is connected to both {targetname1} and {targetname2}. No transformation can be pushed to more than one target.

The transformation {transformationname} cannot be pushed to the target database because it is connected to multiple targets: {targetname1}, {targetname2}

- Provide as much information as possible.
- Be concise and direct.
- Do not start a message with a parameter.